Distributed Generation of NASA Earth Science Data Products

Bruce R. Barkstrom[1], Thomas H. Hinke[2], Shradha Gavali[3], Warren Smith[4], William J. Seufzer[1], Chaumin Hu[3], David E. Cordner[1]

[1]NASA Langley Research Center
[2]NASA Ames Research Center
[3]Advanced Management Technology Inc.
NASA Ames Research Center
[4]Computer Science Corporation
NASA Ames Research Center

NAS Technical Report NAS-05-006
May 2005

Abstract

The objective of this work is the development of grid-based approaches through which NASA data centers can become active participants in serving data users by transforming archived data into the specific form needed by the user. This approach involves generating custom data products from data stored in multiple NASA data centers. We describe a prototype developed to explore how grid technology can facilitate this multi-center product generation. Our initial example of a custom data product is phenomena-based subsetting. This example involves production of a subset of a large collection of data based on the subset's association with some phenomena, such as a mesoscale convective system (severe storm) or a hurricane. We demonstrate that this subsetting can be performed on data located at a single data center or at multiple data centers. We also describe a system that performed customized data product generation using a combination of commodity processors deployed at a NASA data center, grid technology to access these processors, and data mining software that intelligently selects where to perform processing based on data location and availability of compute resources. This demonstration also suggests that we could create a catalog of phenomena related data at multiple data centers, in which the catalog can contain references to the original data in different locations. The catalog is important to providing other users with efficient access to the data belonging to the identified phenomenon.

Keywords: Grid Mining, Product Generation, Subsetting, Phenomena-based Subsetting

---

[3,4] Work supported by the NASA Advanced Supercomputing Division under Task Order A61812D (ITOP Contract DTTS59-D-00437/TO #A61812D) with Advanced Management Technology Incorporated (AMTI).

Author Contact Information

Bruce R. Barkstrom
NASA Langley Research Center
MS157D
Hampton, VA 23681-2199
(757) 864-5676, [FAX] (757) 864-9882
Bruce.R.Barkstrom@nasa.gov

Thomas H. Hinke
NASA Ames Research Center
Mail Stop 258-5
Moffett Field, CA 94035
(650) 604-3662, [FAX] (650) 604-4377
Thomas.H.Hinke@nasa.gov

Shradha Gavali
NASA Ames Research Center
Mail Stop 258-6
Moffett Field, CA 94035
(650) 604-1843
shradha@nas.nasa.gov

Warren Smith
NASA Ames Research Center
Mail Stop 258-6
Moffett Field, CA 94035
(650) 604-0521
wwsmith@nas.nasa.gov

William J. Seufzer
NASA Langley Research Center
MS 125
Hampton, VA 23681-2199
(757) 864-9014
Wiliam.J.Seufzer@nasa.gov

Chaumin Hu
NASA Ames Research Center
Mail Stop 258-6
Moffett Field, CA 94035
(650) 604-4451
chaumin@nas.nasa.gov

David E. Cordner
NASA Langley Research Center
MS157D
Hampton, VA 23681-2199
(757) 864-7325
David.E.Cordner@nasa.gov

1. Introduction

One of NASA's historic missions has been to develop and deploy satellite-borne instruments for observing the Earth. As the volume of data collected by these instruments has grown, NASA has massively expanded access to their data by creating data centers charged with providing search tools and distribution capabilities. In some cases, the centers also produce data using algorithms supplied by the instrument science teams. These centers currently distribute many Terabytes per month to a wide variety of users, usually without charge except for costs associated with reimbursement for media distribution.

By design, these centers store their data in files. Users order complete files that are delivered by media or over the net. Within the archives, the files are organized in directories by time of observation rather than by physical location on the earth. This data organization is cost-effective for producing large volumes of higher-level data products, which are created by applying algorithms to the raw data to create products that represent some geophysical parameter such as ice or sea surface temperature.

The original design of this system also attempted to dictate the use of a single format, known as HDF-EOS, for all of the data in these centers. HDF is Hierarchical Data Format and EOS stands for the Earth Observing System; HDF-EOS is a specific profile of HDF developed with the intent of easing data use for EOS data. However, experience has shown that both the file organization and the data format do not serve the needs of many members of the user community because users have invested large amounts of time and effort in learning to use tools that support other formats.

These two design features can require a user to download large amounts of unneeded data along with the desired data fields. After downloading the data files, the user must use his compute and storage resources to process these large data sets. For many users, this approach is inefficient in that it requires transporting large amounts of unnecessary data by media or over the net. It also requires the data users to have a substantial degree of data management skills – an assumption that does not describe many of the data system's users.

We seek to address these problems by showing how the archives can use grid technologies to efficiently produce the customized data products that our users require, rather than forcing the users to assume the entire burden of doing so. Our approach consists of co-locating commodity Linux clusters with NASA earth science data archives. With this hardware, the archives can use grid technologies to intelligently generate customized data products that are exactly what the user needs. The co-located clusters have high-bandwidth connections to the tape storage units and attached disks in the archives, which speeds data retrieval. In addition, many of the user needs for customized data products only require computations on a "one CPU per file" basis. In other words, the load on the CPU's does not require even symmetric multi-processor capability – the computations are independently parallel. Finally, it is easy to use grid technologies to flexibly reschedule the jobs to be done. This means that Linux clusters and grid technologies are well adapted to serve the user needs for quickly retrieving data from the archives and for producing customized data products that decrease network bandwidth requirements of the centers.

Grid technologies can also help solve a second set of problems that arise from the fact that simultaneous, collocated observations by different instruments may be located at different centers. Currently, a data user who wants a data product derived from data stored in multiple archives must order large amounts of data from each archive. Then, he has to combine this data in his own location. Our approach to this situation is to subset archived data in clusters located in or near the original archive and then intelligently select compute resources which can combine the subsets into the data product requested by the user.

Such multi-center subsetting can be enhanced by having the subsetting process create joint catalogs of phenomena and the data associated with them. In other words, when the subsetting process identifies data associated with a particular phenomenon, the software creates a catalog entry that identifies the phenomenon in question, computes summary properties for that instance, and provides pointers to the data within the files that appear to belong to that phenomenon. These catalogs allow users to identify subsets at different data centers. Because they are small enough to replicate, these catalogs allow users who want to work with phenomena summaries (such as area, average values, and history) to do so by consulting only one data center. Users who want detailed examples of the original data can be assisted by grid technologies that extract the subsets from different data centers and then send these much smaller data collections to the user (or to another resource that then combines them before sending the result on to the user). Such catalogs should make it much easier for data users to undertake interdisciplinary investigations with previously independent data sources. They also increase the efficiency of the community as a whole because one expert group can identify data for a particular collection of phenomena and leave an annotation to instances of the phenomena. These annotations serve as pointers to the underlying data subsets that can then be readily extracted without having to redo the data mining work that identified the original phenomena instances.

In this paper, we first describe the commodity cluster approach to the specific problem of subsetting large data sets. Generally, the subsetting problem arises when there is a large data set that contains data not of direct use to the data center customer. The data center needs to read the original data files, select the portion needed by the user, and then reformat the results to a format selected by the user. To the extent that each file can be subset independently of other files, we can view this subsetting problem as a kind of "data filter" that accepts an input file, operates on it, and creates an output file. This approach is quite general.

After showing how we applied grid technology to the subsetting problem, we describe how we applied the technology to the multi-center problem. In this application, a data mining program at one data center created a subset that identified the data belonging to mesoscale convective systems which was represented by a convex hull polygon. Then, we sent the convex hull polygon surrounding each mesoscale convective system to a second data center that extracted a subset for each convex hull from another data set. Each subset contains data specific to the observed phenomena instances. Furthermore, the separate subsets can be provided to users who can now examine several related types of data describing the same phenomena.

Our preliminary experiences show that our approach can deliver customized data products to users in less time than when the user orders huge volumes of data and generate their own data products. Our experiences also show that grids are a key enabling technology, because they overcome the problems involved in software integration, incompatible software versions, and firewalls.

The next section describes the context within which we perform our work. Section 3 provides a more-detailed description of the problem we are trying to solve, with Section 4 describing our grid-based solution to the problem. Section 5 describes the lessons that we learned in the course of this work. Section 6 presents our conclusions and describes future work with this technology.

2. Background

Our work takes place within the context of the NASA Earth Science community and the Grid community. In this section, we provide information on NASA Earth Science data archives and on the technologies we used, including both data mining, and grid computing. This will set the stage for the presentation of our work.

2.1. NASA Earth Science Archives

Various NASA and non-NASA satellites support instruments that capture remotely sensed data about the Earth. These data are downlinked from the satellites and transferred to various archives. Then, the archives produce calibrated and validated geophysical parameters that are much more useful than the raw data itself. To support this work, NASA developed eight Distributed Active Archive Centers (DAACs), which are intended to hold and distribute long-term Earth observation data from the Earth Observing System (EOS). The DAACs are a significant component of the Earth Observing System Data and Information System (EOSDIS) [Schwaller 1996].

Since there is little scientific interest in raw data, the data contained in the EOSDIS DAACs are carefully calibrated and validated. Calibration, validation, and production must be performed under the care of science teams – most with more than ten years of algorithm development experience. While calibration efforts are continuous, validation takes from two to four years.

Production of these validated data products involves a wide variety of data types, structures, and processes. The production resources required to produce these products can be significant. For example, the Langley Research Center's Atmospheric Sciences Data Center (ASDC) uses 2/3 million source lines of code (SLOC) in the preparation of one of its data products and 1 million SLOC for another. Producing the calibrated and validated data is a substantially automated process, with workloads ranging from 100 jobs per day to 10,000 or more jobs per day.

The DAACs currently have about 3 to 4 petabyes of data and provide data to more than 100,000 customers per year. They distribute many terabytes per week, both via FTP and media. Currently data within these archives can be ordered through various web-gateways spread around the world. These are listed on http://redhook.gsfc.nasa.gov/~imswww/pub/imswelcome/imswwwsites.html. The smallest orderable quantity of data is called a granule. A granule can vary from a file of one or more images to a file consisting of a whole day or entire month of data that incorporates data from many orbits.

These archives were designed over 10 years ago to hold various Earth science data products and to help users to rapidly identify and order desired data products. As Figure 1 shows, the current access pattern is for users to order the data through a web site. The data centers deliver the ordered products either via an FTP (file transfer protocol) site or via media. FTP is the method of choice if the order is small (less than a few gigabytes). Tape or CDROM delivered via the postal service is used for larger data orders. Currently, the actual extraction of the ordered data is under manual control since most of the data is stored on tertiary storage and must be retrieved from the mass storage system. The archives were not designed to support direct access by users to the mass storage system, primarily because the number of tape drives limits the capacity. This means that access must be controlled since these drives could not support arbitrarily large data requests from the web community. This community is quite large – more than two million distinct IP addresses accessed the web interfaces of the EOSDIS data centers last year.

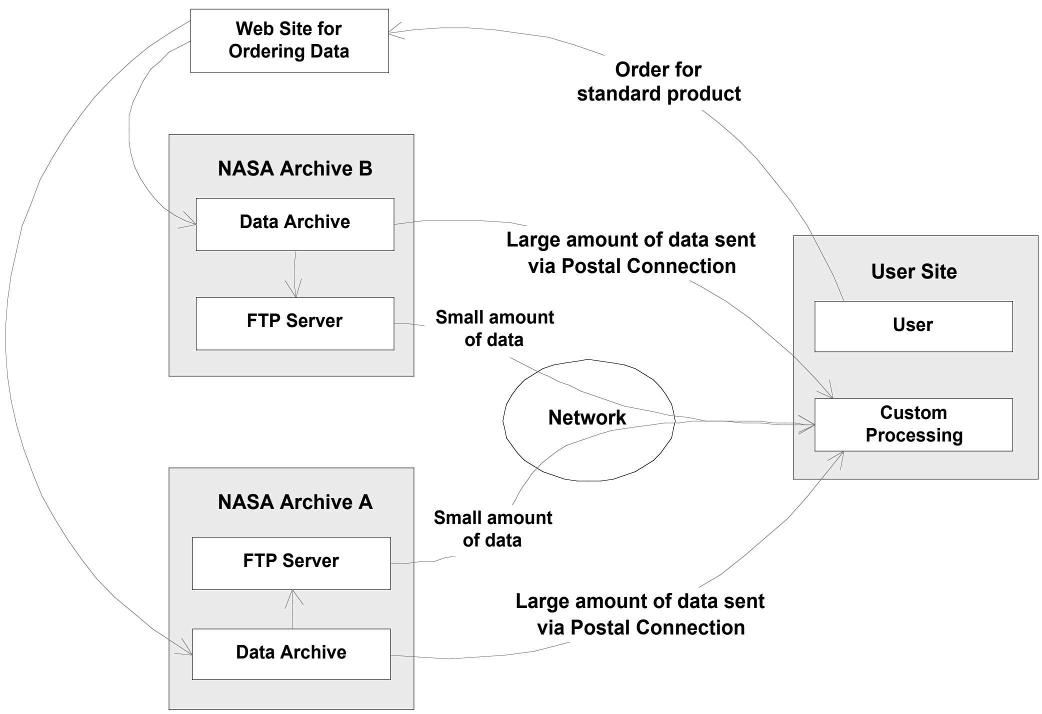

Figure 1. Current practice to retrieve EOSDIS data and produce a customized data product.

2.2. Data Mining

The NASA Workshop on Issues in the Application of Data Mining to Scientific Data that was held in October 1999 had a presentation on grid mining by one of the authors of this paper [Mining Workshop 1999]. This work, which is briefly described in [Hinke 2000b] is the first known description of data mining on a grid. The work by Du and Agrawal using DataCutter is a recent example of grid mining [Du and Agrawal 2002]. To date there has not been much work on grid-based data mining. The workshop on Data Mining and Exploration Middleware for Distributed and Grid Computing, while having a broader focus than just grid-based mining did provide a forum for reporting on recent work on grid-based mining. [Mining Middleware 2003] While much of the work reported upon at the workshop could have applicability to data mining on a grid, a number of projects reported upon at the workshop specifically addressed grid-based data mining, including the United Kingdom's e-Science Discovery Net, the Italian work at the University of Calabria on grid-based mining within the Knowledge Grid framework, Agrawal's grid-based data mining work at Ohio State University, and the NASA Grid Miner work. The conclusions to date are that while grids are certainly being considered for and in some cases used for limited data mining, the coupling of data mining and grid technology is still in its infancy.

2.3. Grid Computing

Over the past several years, researchers in government laboratories and academia have developed and deployed hardware and software to support large-scale distributed systems. We call these systems computational grids, or just grids [Foster and Kesselman 1999]. The purpose of such grids is to enable scientific discovery by allowing users to perform faster analyses, larger scientific simulations, or access and manipulate larger repositories of scientific data than were previously possible.

The NASA Information Power Grid (IPG) is a project to research, develop, integrate, and deploy grid technologies at NASA centers [Johnston, et al. 1999]. The purpose of this project is to create grid environments that allow NASA scientists to better perform their research. For much of its history, the IPG has focused on enabling the execution of applications on grids so as to reduce the time required to perform scientific processing for such applications as simulations or parameter studies. Recently, the focus of the IPG has shifted to enabling access to and processing of the large amounts of scientific data that NASA gathers and archives. This work makes use of software developed and deployed by the IPG, as well as compute and storage resources.

The work described in this paper is similar to the data-intensive science grids that are being developed such as Grid Physics Network (GriPhyN) [Avery and Foster 2000], the Particle Physics Data Grid (PPDG) [PPDG], the EU Data Grid [Ghiselli 2002], and the National Virtual Observatory (NVO) [NVO]. These projects collect data from various instruments, store this data in distributed archives, and then provide this data in raw or processed forms to users. These projects target new data-intensive science grids and spend a large amount of effort developing general data management systems [Baru, et al. 1998, Chervenak et al. 2001, Guy, et al. 2002] and general virtual data systems for transparent creation of data products [Foster, et al. 2003]. In this work, we have a legacy system that has custom data management tools already in place. Thus, while we find the data-intensive grid approach interesting, we have not yet required a general virtual data framework for creating the customized data products that are particularly useful to the data centers customers.

3. Producing Customized Data Products

One of the implications of the processing pattern shown in Figure 1 is that all of the data needs to be delivered to the user's site for the custom processing required by the user's research. In other words, the user must retrieve all of the granules from the archive(s) that contain data that may be relevant to the user's research. This approach often means that the user retrieves much more data than he really needs.

Consider the case where the user is doing a ten-year study of some phenomena in California. If the data of interest were packaged in granules that include all of the data from all of the orbits for a single day, then each granule would include the desired data from California (a small part of one or two orbits), as well as the data for the rest of the world. This means that the user would be retrieving ten years of global data, which then has to be run through software that will ignore most of it except for the small portion that covers California.

One solution to this problem is for the archive to send the user only the data that he needs for his research. This solution requires software that can produce a subset of the total data for the area and period of interest to send to the user. More generally, we can describe this approach as a subsetting service provided by the archive. Note that subsetting is probably the simplest example of an application that can be used to support the creation of custom products at the archive. Other applications, such as reformatting or data mining itself could also be moved to the archive to reduce the amount of data management or data processing that has to be accomplished at the user's site. We describe a grid-based approach for accomplishing subsetting in the following sections.

4. Approach

This work has used a two-pronged solution to the creation of custom data products. The first branch of this solution is to move subsetting to the archive and provide a computational infrastrucure that can support user specified subsetting. The second branch is to develop software that allows a specification of a subset from one data center to control the generation of a collocated data set at another center. This second branch is particularly interesting because it addresses how to create custom data products that rely on data from multiple archives to efficiently work across a distributed system.

4.1. Move Processing to Archive

Moving processing – in this case subsetting – to the archive is the first part of our solution for providing custom data products. Subsetting at the archive reduces the amount of data that must flow between the archive and the user. While various technologies could have been used to support this archive-based processing, we selected grid technology. The primary reason for this selection is that there are multiple Earth science archives that may be involved in creating custom data products. In this case, the technology selected needs to enable forming a distributed system that crosses administrative boundaries. In other words, the data needed by the user falls under the jurisdiction of multiple archives, some belonging to NASA under the administrative control of different NASA centers, while others may involve non-NASA organizations.

Grid technology is designed for just such an environment, making it the clear choice for our work.

However, even for "content-based" searches or subsetting within a data center, grid technologies can be advantageous with large-scale subsetting. Figure 2 shows a processing pattern that the grid needs to support to enable production that accesses several sources of relevant data within one archive. With this approach, the archive provides parallel custom processing and sends to the user only the data that is directly relevant for the user's research. In the previous example, the archive could subset a 10-years data set of whole-Earth data to extract only the portion that covers California. Then, only this subset would be sent to the user.

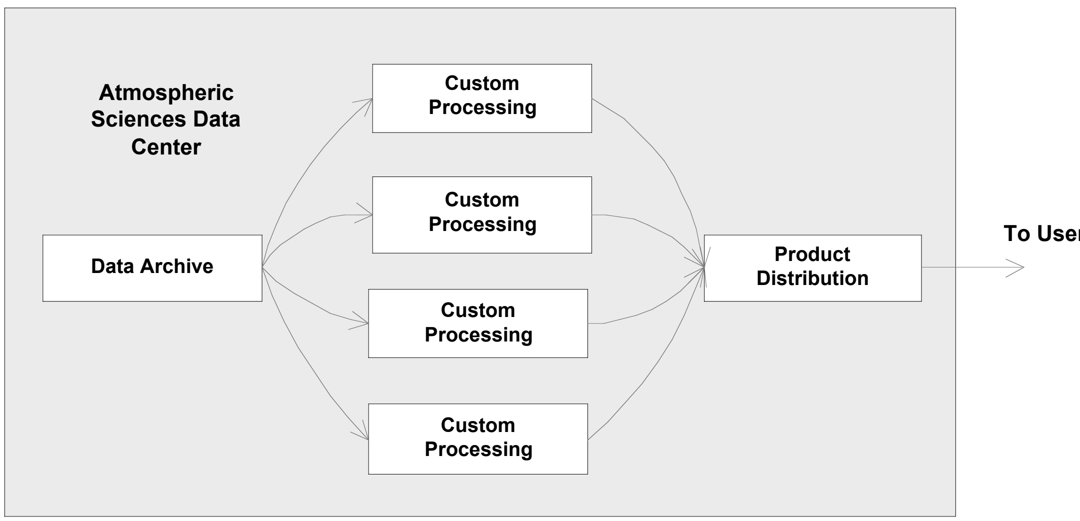

Figure 2. Intracenter parallel processing.

Grid computing technology appears to offer substantial advantages for the kinds of data searches and customized processing we have just discussed. First, even though this processing pattern reduces the volume of data that needs to be transported across the network, the data volumes and number of files being sent from the data center to the user is still large enough to benefit substantially from the reliable FTP services of grid protocols. Second, in more advanced interdisciplinary research, grid computing aids coordination of processing tasks that would be carried out in different data centers. Third, some very useful tools such as the Storage Resource Broker (SRB) [Wan 2003], permit grid-based applications to retrieve data directly from tape silos. The SRB can translate requests for data known to the SRB to the underlying mass storage system that provides access to the tape storage silo. In his way, the SRB provides a seamless access environment that frees the grid-resident application from having to deal with issues of file staging from mass storage to disk and from disk to the particular grid processor that performs the subsetting operation.

The initial subsetting application of most interest to ASDC is subsetting data that is not image-like. Many of the data products in the center are best thought of as one-dimensional arrays of multi-parameter records, where the time of observation forms the index to the arrays. Other data consists of time-ordered vertical profiles of chemical composition or equal-area grids of monthly average, multi-parameter records. For this initial example, we concentrated on subsetting data products where each granule contains an hour of data dealing with cloud properties and radiative fluxes from the instruments of the investigation of Clouds and the Earth's Radiant Energy System (CERES). The CERES scanner is a non-imaging instrument with a complex spatial sampling pattern.

Because the data of interest are not image-like, the subsetting operation is similar to a database query involving selections of records based on the values of the parameters in the records. This approach also allows a user to examine relationships between parameters for the selected records, thereby bringing out more interesting information than would be possible by examining the relationships formed by unselected data. The important point for this work is that the sought-for relationships exist over many selection instances – and often require many selected samples to reduce the noise in the relationship. As a result, climatological data users need to examine a very large amount of data to create meaningful and certain results.

In terms of the data center, this kind of interdisciplinary climatological data use creates requirements for streaming large files through filtering programs at a very high rate. For example, the CERES files for this example include one hour of data. A climatological investigation that needed to work with a six-year record from one of the CERES scanners on the Terra spacecraft and a six-year record from a CERES scanner on the Aqua satellite would have to subset about 105,000 files. These files contain about 21 TB of processed data.

Research on the data recorded in such file collections is impeded (or, more likely, impossible to support) if examination of the entire data record is likely to take more than a few months. If data access slows beyond this interval, it is essentially impossible for a user to create validated production algorithms because he or she will not have sufficient time to correct errors. The data access rate has to be high enough to allow iterative experimentation. High data access rates are also needed to support both migration of large data collections and a moderate number of climatological data users who need reasonable turnaround (say 1 week or less) in working with five or ten years data. Assuming that the major bottleneck lies in the I/O, we may need to do some additional architectural work on how we store the files, but this will await experience with the system being described in this paper.

Perhaps the simplest way of describing the systems currently used by the ASDC is that they were engineered to deal with data storage – under the assumption that data users want only a few files in each access and that the accesses are more or less randomly distributed across all of the files. This access pattern makes for a reasonable match between robotic tape storage, the data storage strategy, and the user access pattern. However, this match breaks down in the face of the need for content-based data subsetting. In dealing with this problem, we currently have no perceived need to worry about parallelizing the individual program code – computation isn't the bottleneck, data bandwidth is.

That fact almost certainly means that the new architecture needs to spread the data over multiple disks. Then, the application needs to run it through as many CPU's as we can make available, using coarse-grained parallel processing, where each processor can work on data for a different day or orbit. This notion gains support from the performance of the subsetting program we describe in more detail below. Performance statistics show that 90% of the total execution time of the test case was devoted to opening the file and reading 10% of the one-dimensional arrays in the file. Nine percent of the execution time was spent on writing an ASCII output file. Only one percent of the execution time could be allocated to "computation", assigning this performance to the computations that select the data that belong in the final subset. While these performance diagnostics can vary over a large range, depending on both the data being examined and on the selected operations the user wants to perform on it, they suggest that the performance of the subsetting program is dominated by I/O, rather than the kind of computations that typically appear in computational modeling and simulation.

In other words, from a system perspective, the problem is not just increasing the total throughput, but managing the overall flow of data from storage nodes to CPU's and back to storage. Grid technologies are particularly useful for such management. Both storage management and job scheduling are key components of grid technology.

On the technology side, the required throughput can be achieved by transferring the data from tape silos to a large number of disks, with an appropriate number of CPU's to perform the data filtering and subsetting operations. To move toward this goal, ASDC transferred a modest number of files out of the archive for storage onto disk. Then, the center provided a simple subsetting program that allowed users to interactively build a script of instructions similar to what one might do with a relational database. The intent is to allow this program to create a data structure similar to the rows with fields in a database and then to perform the following four basic functions on the data:

- Use simple queries on the fields to select rows into the subset
- Calculate simple statistics on the fields in the selected rows
- Visualize the relationships between the fields – at high display rates that would allow millions of data points to be plotted on a user's browser in under thirty seconds
- Create transformed variables that are placed in new columns of the in-memory data structure

Figure 3 shows an architectural view of the ASDC's grid-based data access and production system and Table 1 describes the systems used and their function. This system will initially support parallel subsetting, but could just as easily support parallel reformatting of the data to transform it from its archived format into a format desired by the data requestor. It could also perform other processing to customize the data for specific user needs. It is expected that both internal and external data users will interact with this system through a web-services style of interface. At the same time, it is important for the system to be sufficiently automated so that intelligent agents could provide files for ingesting data and for obtaining data through the distribution interface – expected to be push and pull FTP. Internally, both the CPUs and the storage nodes should be designed as peer-to-peer daemons to increase system reliability and to improve security.

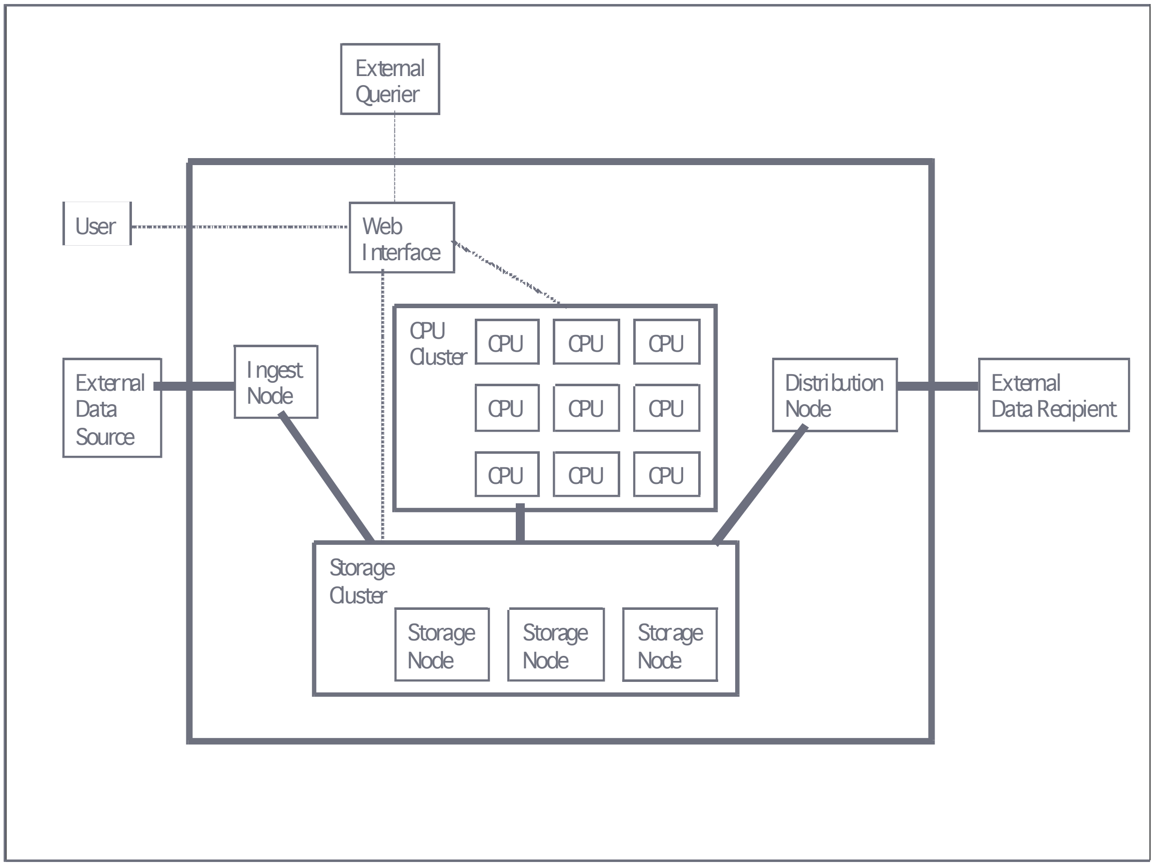

Figure 3. A Schematic Architecture of Grid-Enabled Data Access and Production System. Heavy lines show high-bandwidth data transfer paths; lighter (dotted) lines show command paths. Both CPUs and Storage Nodes are intended to operate autonomously to increase the reliability and security of this architecture.

System Name	Operating System	Location	Use
rime1	Linux	Langley Research Center	Started scripts on rime1, rime2, rime3 and rime4 to perform parallel subsetting, This machine performed subsetting on 30 files
rime2	Linux	Langley Research Center	Performed subsetting on 30 files
rime3	Linux	Langley Research Center	Performed subsetting on 30 files
rime4	Linux	Langley Research Center	Performed subsetting on 30 files

Table 1. Systems Used for Parallel Test

For the work reported upon in this paper, the XML-driven subsetter developed by the ASDC ran on a (Globus) grid-enabled Linux cluster, first reading 120 files (five days) of CERES SSF (Single Scanner Footprint) data, using sequential job runs on a VA Linux machine (rime2), and then concurrently on four CPU's of the cluster (rime1, rime2, rime3, and rime4). In this system, the subsetter reads the XML template file, which identifies the data file to be read and the location of the subset file when the job is finished. The XML template file identifies which parameters to read from the file and is intended to allow four kinds of operations to be performed iteratively and interactively on the data in a given file.

The incoming data from CERES SSF contain 131 parameters, organized in a singly dimensioned array, typically with about 150,000 independent observations, or footprints. The XML template allows a user to specify an arbitrary fraction of these parameters. For purposes of this test, 14 parameters were selected. The template then intended to allow users to specify whether they want to select footprints within a particular range of values, produce a statistical summary, compute a new parameter from the old set, or visualize relationships. For purposes of this test, the template selected all footprints within the range of 100 W/m^2 to 220 W/m^2 for Long-Wave Flux, where this range should select all footprints belonging to cold, thick clouds. In the test, the number of footprints in the selected subset was typically about 35,000. Between the ten percent selection of the original variables and the reduction to one-fifth of the original number of footprints, this subsetting program was effective in reducing the number of data elements available for delivery to the user to about $1/50^{th}$ of the original number.

To compare the performance of a grid-resident serial version of the subsetter, with a grid-resident parallel version, a test was run comparing the performance of a single processor with a four processor subsetter. The serial version ran for 948 seconds against 5 days of CERES SSF data. Using four processors, the subsetter ran for 305 seconds, which was a speedup slightly better than 75% of a linear speedup.

The following is a description of the grid environment in which these tests were run:

- Globus 3.0 alpha
- GridFTP [GridFTP 2002] for all file transfers.
- Secondary storage (local disk) to store the Globus jobs (shell scripts
- Tertiary storage (disk on network attached computer - rime1) for data files and subsetter binary

The procedure for the serial test included the following:

- Ran on rime2 (submitted from rime1)
- Script started a Globus job on rime2 to GridFTP data and subsetter to rime2
- Waited for all data to arrive
- Started timing
- Started a globus job on rime2 to run subsetter against 5 days (120 files) of data
- Finished timing
- Saved logs and data output

The procedure for the parallel test included the following:

- Started a script from rime1 that started a globus job on rime1, rime2, rime3, and rime4 to GridFTP data and subsetter to each of the machines (30 on each – for a total of 120 files, the exact same files used in the serial test)
- Waited for data to arrive on all 4 machines

- Started timing
- Started a script from rime1 that started a globus job on rime1, rime2, rime3, and rime4 to run subsetter against 30 files of data each for a total of 120 (5 days) across the system
- Finished timing
- Saved logs and data output

The overall approach that has been described follows the recommendations in the recent report of the National Research Council on Government Data Centers [NRC 2003], which recommends that data centers consider moving from tape storage to disk and incorporating more "bleeding edge" technologies. We note that recent advances in versioning theory for Earth science data products [Barkstrom 2003] suggest a clean separation between data in files and metadata that describes file collections. This work provides data structures that give the system the ability to perform provenance tracking from "cradle to grave" on the data that stored in this system. These features also accord well with the recommendations of the NRC.

4.2. Create Inter-Archive Data Products

An approach for combining data from multiple archives (in this case two) is the second part of our overall solution for providing custom data products. In this case, the objective is to reduce the amount of data that must flow between archives in order to produce a desired custom product. While the intent of the research was not to investigate a comprehensive solution to this problem, the goal was to look at an innovative approach for a significant type of multi-archive data product that would be of use to the scientific community.

The second processing pattern to be described in this paper involves processing of data archived at multiple archives. It seeks to avoid moving relevant data from multiple archives to a single site. Rather, it seeks to "parallelize" the processing across multiple archives, while still making the results visible at several. Figure 4 shows this pattern, using the example of having one archive mine a data collection and then send a set of instructions on how to locate the equivalent data to the second archive for custom processing there.

An example of this pattern derives from a user's desire to investigate the properties of mesoscale convective systems (severe storms) that have been simultaneously observed in multiple data sets. Under this processing pattern, a microwave data set in one archive can be mined for the location and microwave emission patterns of the mesoscale convective systems. The mining process creates an index of times and locations that can be used by the second archive to extract the observations of cloud properties and radiative fluxes obtained by very different kinds of instruments on the same satellites. The data centers can act independently to provide the microwave subset and the cloud and radiation subset to the user in a way that makes it easy for the user to combine the results for a much more interdisciplinary examination of the severe storm phenomena. This approach gains two important kinds of efficiency:

- The data centers do not need to deal with exporting or importing very large volumes of data;
- The user does not need to manage data in which he is not interested.

Again, since this approach involves distributed processing that crosses administrative boundaries, grid technology is highly attractive.

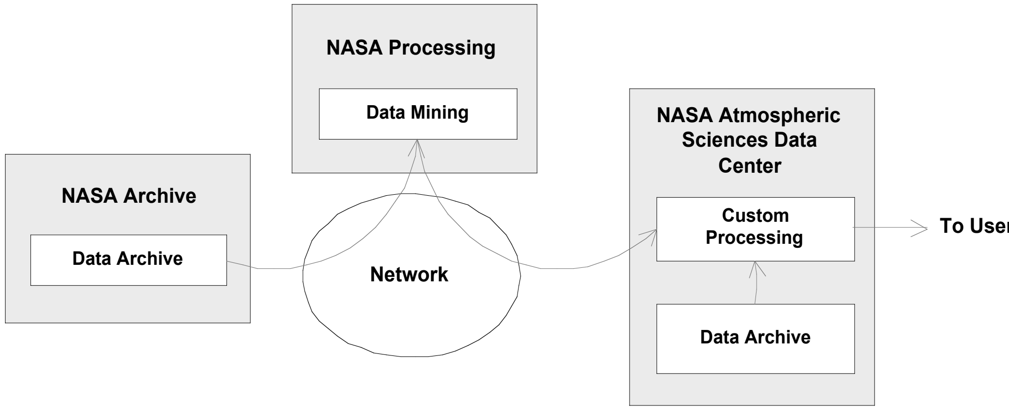

Figure 4. Multi-archive collective processing.

In a sense, the two patterns we have been discussing may be characterized as streaming data processing on very large volumes of data (processing pattern 1) and content-based data searches for phenomena of interest (processing pattern 2), The current NASA archives have not been designed to support this type of processing, although it would be very useful for supporting interdisciplinary research needed for climate understanding. There has been some experimentation with subsetting, particularly for applications involving geospatial searching for data found in the vicinity of fixed Earth locations or regions. Thus, there are a number of tools that appear in the search-and-order interfaces of the NASA data centers. These tools can assist users in finding granules near a given location and can then extract only those portions that satisfy the relevant “nearness criterion” for geospatial searching. However, there have been no large-scale attempts to provide a content-based search for the very large collections that appear in these archives.

Since related data on a particular phenomenon may be located in different data centers, we anticipate the need to help users bring together correlated subsets from multiple locations. To minimize the resultant data flow, we would be better off exchanging "pointers" or feature indexes to the data rather than entire data sets. Such “pointers” allow features discovered in one data center to be used to subset data in another data center. This approach should reduce the required network traffic by several orders of magnitude over an approach in which data from multiple centers is all moved to the same location to perform multi-data-set fusion. It also allows data centers to retain autonomy with respect to their collections at the same time it increases services to users. The feature indexes serve as an additional source of metadata on the phenomena themselves. Thus, the pointers can help new user communities find objects of interest can be replicated at both data centers for a very small storage overhead. We expand on this approach in the section that follows.

Phenomena-based subsetting is a concept that supports the desire to perform research on data from a number of different datasets that are all associated with the same phenomena. In order to support phenomena-based subsetting, the spatial and temporal location of the phenomena of interest must be determined. Since finding these locations could involve sifting through a large amount of data to locate phenomena of interest, it represents a potentially good application for data mining, which has been defined as "… the process by which information and knowledge are extracted from a potentially large volume of data using techniques that go beyond a simple search through the data" [Data Mining Workshop 1999]. Scientific data mining in general, and Earth Science data mining in particular is characterized by the need to mine large amounts of data that has been captured by satellite-based remote sensors. An example is data from the TMI (TRMM Microwave Imager) instrument, which consists of approximately 230 megabytes (uncompressed) of data per day. Other satellite data can be even more voluminous, particularly if the resolution is finer than the TMI data.

This part of our research uses a software system called the Grid Miner [Hinke 2000b] that was developed at the NASA Ames Research Center. The Grid Miner is a grid-enabled version of the stand-alone ADaM data mining system that was developed at the University of Alabama in Huntsville under a NASA research grant [Hinke 2000a, Hinke 1997a]. The Grid Miner is an agent-based mining system in which mining agents are sent to processors on the grid to mine remote data that is accessible from the grid and described in a mining database that has been pre-loaded with the URLs of data to be mined.

Figure 5 shows the architecture that is used to perform phenomena-based subsetting. A user invokes the miner by staging "thin" mining agents to the grid processors that are to support the mining, along with the mining plan (written by the user) that is to guide the mining for the desired phenomena. Based on the mining plan provided, these thin agents are able to grow in capability through the acquisition of the necessary mining operations required to execute the plan. Each of the mining operations is configured as a shared library executable, with one operation per executable file. As the thin mining agent executes the mining plan, it identifies the operations that are to be used and then uses the grid to transfer the needed shared library executables from a mining operator repository to the grid processors where the mining is to be performed. The use of thin agents minimizes the size of the agent code that needs to be transferred. The approach of dynamically acquiring needed mining operations means that mining operations could be retrieved from multiple operator repositories, some public, some private and perhaps some for a fee, although this multi-site repository represents future work.

Once the thin agent has grown to have the necessary mining operations to perform the mining plan, the mining agent contacts the mining database to acquire the URLs of the files to be mined. Using the grid, these files are then transferred to the mining site and the mining is performed as specified in the mining plan. While Figure 5 shows the basic approach to give the reader an overview of the basic approach, the actual details of the system, shown in Figure 6, are more complex.

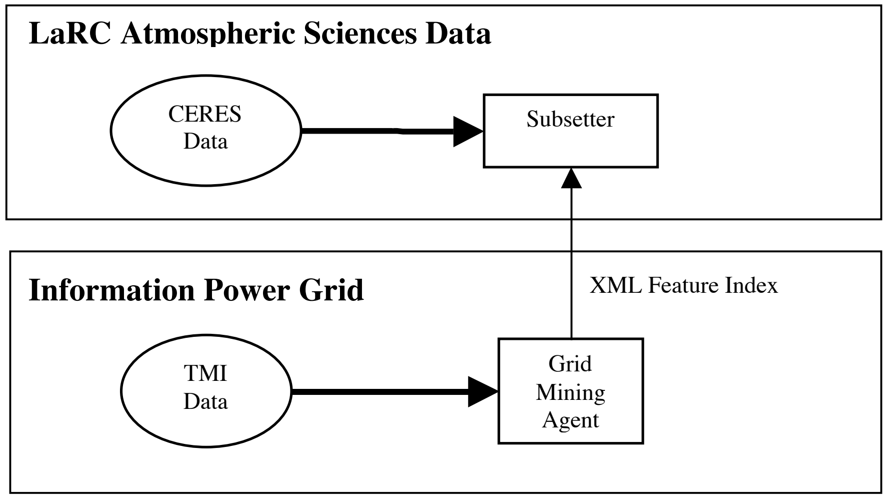

Figure 5. Overview of Architecture of Phenomena-based Subsetting. Heavy lines show high-bandwidth data transfer paths; lighter show the path of the XML feature index. Note that this architecture uses both the IPG and an internal Atmospheric Sciences Data Center Grid.

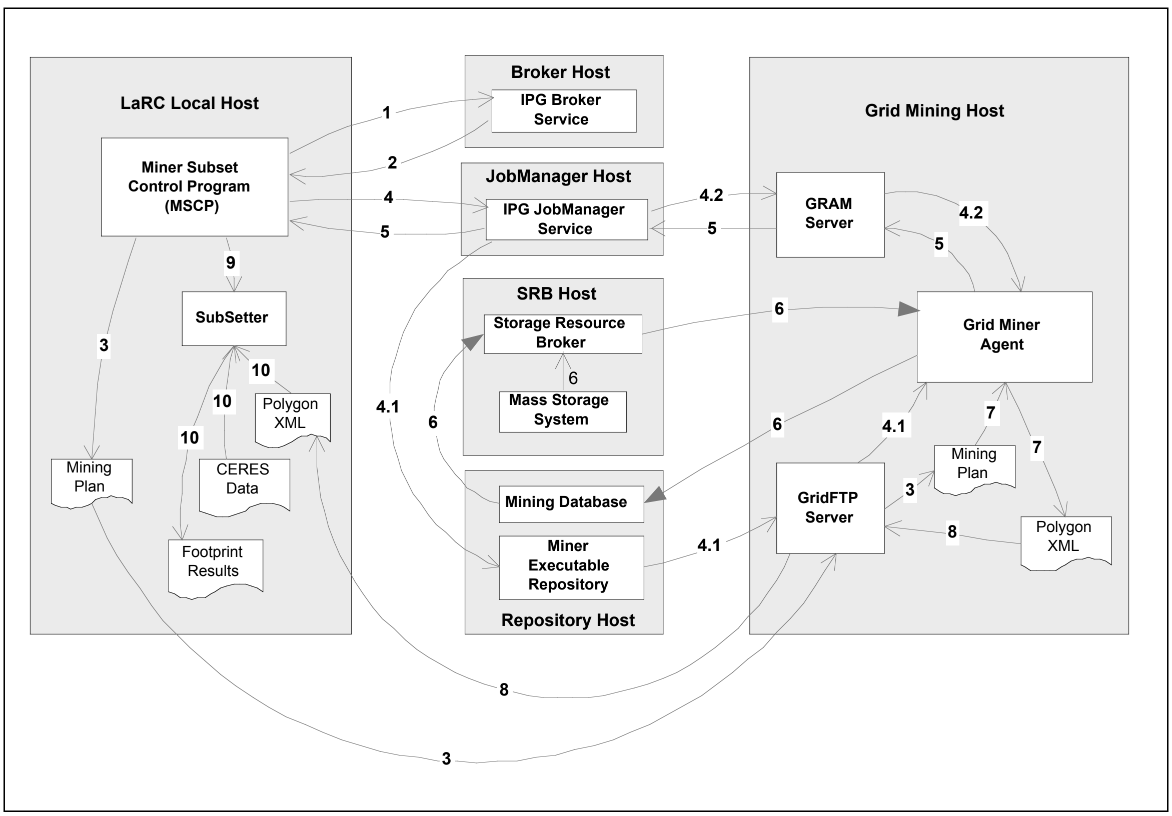

Figure 6. Schematic Diagram for Architecture of Phenomena-based Subsetting (Explanation follows in text).

The following describes the set of operations performed in the course of supporting phenomena-based mining:

1. MSCP (Miner-Subsetter Control Program) passes a set of resource specifications (i.e. the type of the operating system, the maximum amount of physical memory needed, the number of required CPUs, etc.) and requests the IPG Broker Service, which runs on the Globus Java CoG, for selecting and suggesting a list of available IPG resources. [Laszewski 2000, Cog 2003]
2. Based on the specifications from MSCP, the IPG Broker Service selects and returns a list of IPG resources. After receiving the list, the MSCP chooses a system among the list to be used for the Grid Mining Host. If the selection criteria are comprehensive, then the selection can be random, since any suggested system will support the desired mining.
3. MSCP uses a "put" method provided by the GridFTP of the Globus Java CoG to transfer the Mining Plan from the LaRC (Langley Research Center) Local Host to the Grid Mining Host.
4. MSCP uses the IPG Job Manager Client API to construct and submit a job to the IPG Job Manager Service. The job contains the requests to transfer (4.1) and execute (4.2) the Grid Miner Agent executable on the Grid Mining Host.
   4.1. IPG Job Manager Service uses the 3rd party transfer capability provided by the GridFTP of the Globus Java CoG to transfer the Grid Miner Agent executable from a Grid Miner Executable Repository to the Grid Mining Host.
   4.2. IPG Job Manager Service uses the GRAM (Globus Resource Allocation Manager) of the Globus Java CoG to kick off the execution of the Grid Miner Agent executable.
5. The Job Manager Service keeps MSCP informed for any status update of the execution of the Grid Miner Agent.
6. Grid Miner Agent uses the URL addresses provided by the specified Mining Database to access the HDF files located in the Mass Storage System, through the SRB (Storage Resource Broker).
7. Grid Miner Agent performs the mining operations with the input of the Mining Plan and the HDF files and then generates a polygon XML file.
8. When the MSCP was informed by the IPG Job Manager Service that the mining execution is completed, the MSCP uses the "get" method provide by the GridFTP of the Globus Java CoG to transfer the polygon XML file from the Grid Miner Host to the LaRC Local Host where the MSCP is running.
9. MSCP kicks off the Subsetter executable on the LaRC Local Host.
10. Subsetter executable uses the polygon XML and CERES data files as input to perform the subsetting operations and then generates the CERES footprint result files.

To support phenomena-based subsetting, the Grid Miner will mine the data for the desired phenomena and when found, will circumscribe the phenomena with a convex hull polygon and associated metadata to specify not only the spatial extent of the phenomena, but also its temporal location. These will be output as an XML document. A portion of one XML document produced by the Grid Miner is in Listing 1.


```
<polygon>
<julian_date_time> 2450904.754815 </julian_date_time>
<human_date_time> 1998-04-01 GMT 06:06:56
</human_date_time>
<size_in_square_km> 2083.126221 </size_in_square_km>
<region_type> 2 </region_type>
<vertices>
<number_of_vertices> 64 </number_of_vertices>
<vertex>
<latitude>   -2.26 </latitude>
<longitude> -178.28 </longitude>
</vertex>
<vertex>
<latitude>   -2.08 </latitude>
<longitude> -178.38 </longitude>
</vertex>
.
.
.
</polygon>
</polygon_list>
```


Listing 1. Portion of Grid-Miner-produced XML document for two of the 64 vertices that comprise the convext hull polygon for the third mesoscale convective system found in the TMI data for April 1, 1998.

For the initial phenomena sought for this work, we looked for mesoscale convective systems within passive microwave data from the TMI (TRMM Microwave Imager) instrument that were originally obtained from the Goddard Space Flight Center's Distributed Active Archive Center, but was then moved to a grid-connected system at the NASA Ames Research Center for the purpose of supporting this work. The mining operation used to search for mesoscale convective systems was developed at the University of Alabama in Huntsville and uses an algorithm originally suggested in [Devlin 1995]. For the purposes of this experimental work, the TMI data is being kept on storage that is accessible from the NASA Information Power Grid (IPG). The Grid Miner is staged to IPG computational resources, which could be located anywhere on the IPG, with the data to be mined pulled from Ames' storage.

When the mining is completed, the XML document describing the spatial and temporal location of the phenomena of interest was sent to the subsetting engine. Again, the use of XML for transferring information accords well with the NRC recommendations [NRC 2003]. The XML document includes indices to the original pixels in the data files that contribute to the object identifications. Such object indices provide the ability to efficiently retrieve the original data belonging to the phenomenon, as well as building metadata for each instance. This approach allows us to develop a database of phenomena instances [Hinke 1997b] that should markedly increase the scientific community's ability to extend the value of its data resources, as suggested several years ago [Barkstrom 1998].

For this initial work, based on the date and time information provided in the XML document, the appropriate CERES data was accessed and fed into the subsetter, along with the polygon that describes the spatial and temporal areas that correspond to the mesoscale convective systems from the TMI data. The subsetter then extracted the CERES SSF (Single Scanner Footprint) data that corresponded to the TMI-discovered mesoscale convective system. Listing 2 shows the 9 CERES SSF footprints discovered in the third convex hull discovered in the TMI data for April 1,1998:

```
Convex Hull:  3 with  9 footprints
   Footprint:      1      15804
   Footprint:      2      15805
   Footprint:      3      16090
   Footprint:      4      16091
   Footprint:      5      16094
   Footprint:      6      16376
   Footprint:      7      16377
   Footprint:      8      16381
   Footprint:      9      16382
```

Listing 2. CERES SSF footprints for April 1, 1998 hour 6 corresponding to the third mesoscale convective system found by Grid Miner.

In a test involving mining of TMI data from April 1, 1998, the Grid Miner discovered 37 mesoscale convective system polygons in approximately 21 minutes of processing. In less than one second of processing the subsetter identified from 9 to 185 CERES SSF footprints for each of 36 polygons produced by the Grid Miner, out of a total 37 polygons. The test used the systems described in Table 2 that included 6 systems at three NASA Centers.

System Name	**Operating System**	**Location**	**Use**
rime13	Linux	Langley Research Center	Hosts control program to start and control processing, mining plan source, performs subsetting
Evelyn	Irix	Ames Research Center	Mining executable repository, hosts mining database
ipgsrv01	Linux	Ames Research Center	Hosts Broker, Job Manager
Ranma	Sun OS	Ames Research Center	Hosts SRB, Miner LDAP server that points to location of executable repository for specific type of system

Ariel	Irix	Ames Research Center	Mass storage front-end
Sharp	Irix	Glenn Research Center	System used for mining

Table 2. Systems Used

It should be noted that in this case (and by intent) the two instruments (TMI and CERES) were both located on the same satellite – although the data are archived in separate NASA archives. Thus, the subset extracted from the CERES data has both temporal and spatial congruence with the phenomena discovered in the TMI data. The subsetting algorithms described in section 2 were extended for finding all of the CERES footprints to provide a particularly efficient spatial search for the coincident data.

Briefly, both the TMI and the CERES SSF data are ordered along the orbit track by the time of observation. This constraint allows us to replace a three-dimensional search problem (two space dimensions and time) with a one-dimensional search for CERES footprints that might be contained within the convex hulls from TMI. There were 37 convex hulls in a single day's test data. Each hull covered an area of about 2500 km**2, or had a size roughly 50 km by 50 km. Mining 230 megabytes of TMI data on the grid to obtain these 37 convex hull took 21 minutes including transfer of data from Ames Research Center in California to the mining site at Glenn Research Center in Ohio.

Since the CERES data in a single hour includes a spatial span of about 25,000 km along the orbit track, the fraction of an hour's data that might have footprints within a convex hull is about 0.3%, allowing for the fact that several of the hours have pairs of convex hulls that overlap in the along-track direction. Because the search for footprints within the convex hull can ignore portions of the orbit where there are no convex hulls, this approach is very fast. Timing on the grid-enabled Linux cluster showed that identifying all of the CERES footprints within the convex hulls took only about 0.5 seconds for an entire day of data. As with the subsetting for processing pattern 1, in which it typically took about 2 seconds to extract about ten percent of the data and then locate the one-fifth of the records with low long-wave fluxes, the phenomena-based subsetting for pattern 2 is sufficiently fast that the architecture we suggest appears to be capable of meeting the needs for content-based searching and subsetting of very large quantities of data.

5. Lessons Learned

In the experiments we have described in this paper, Grid technology provides a number of useful services that enable much higher data access for users than the current technology:

- GridFTP provides a reliable means of moving data between the various sites, with the grid providing a single-sign-on environment (user identifies and authenticates himself to one grid system, and the grid handles his access to all other grid resources used).
- A grid-enabled storage system, such as the SRB, provides a seamless interface through which the Grid Miner could directly access data stored on a mass storage tape silo with as much ease as accessing data stored in the file system of one of the grid-accessible processors.

- Grid technology provides a single-sign-on environment for running the grid miner on available grid resources and handling the transfer of both mining operators and data.
- The Java-based COG kit provides a powerful capability to easily create a control program that coordinates file transfers and processing in two different grid environments, operating under two different versions of the Globus grid software, GT2 (Globus Toolkit version 2) for the Information Power Grid and GT3 (Globus Toolkit version 3) or the Atmospheric Sciences Data Center grid at Langley Research Center.
- The grid provides a convenient means of transferring feature indexes from the Grid Miner into the data center's subsetting engine.
- The Storage Resource Broker (SRB) provides a useful way of separating the details of the local storage from the logical structure of the files and directories, reducing the operator overhead associated with storage management, thereby reducing the total cost of ownership.

It is important to note that NASA data centers, as with many other locations, are engaged in daily operations that include high-volume data production and large data transfers to users. One of the challenges these centers face is merging technology that has been demonstrated on a small scale into more "mundane" operational environments. While we have not dealt with this challenge in this paper, it is clearly one of the more difficult aspects of evolving these data centers – and requires careful consideration of the total cost of ownership, including upgrading and maintaining the hardware and software involved in "operationalizing" the research approach we discuss here.

6. Conclusions and Future Work

We believe that coupling grid technology to archives provides an exceptionally rich environment for the development of custom data products that have more value to users than the original data has. In some cases, the grid will support processing within the data center on an internal grid, as was the case with the subsetter for this work. In other cases where high speed network connectivity is available, it may make sense to move the processing away from the archive, particularly during periods when the archive's resident computational resources are all being heavily used. Such periods may arise when a particular dataset is being reprocessed with an improved algorithm or when data are moving from an old storage medium to a new one. Of course, if the custom product of interest uses the data from the dataset being reprocessed, then the archive's internal grid-resident processors can be used, not only to reprocess the data, but also to subset the desired data as it is extracted from the mass storage system, reprocessed, and then stored back on the same or a new mass storage system.

Future work needs to be done to investigate the balance between user-specified processing at the archive and subsequent processing at the user's site. Even if all archives are upgraded with the commodity processors used in this work, they may never be able to support all of the processing that users may want to perform at the archives. Thus, work needs to be done to perform significant processing at the archive, but not so much as to overwhelm the archive with custom user processing.

Acknowledgements

We would like to acknowledge the fact that the Grid Miner is a modification of the ADaM data mining system that was developed by the University of Alabama in Huntsville, under a NASA research grant. While Thomas Hinke, one of the authors of this paper designed a data mining system that was a precursor to the ADaM data mining system and was one of the investigators involved in the development of ADaM, Dr. John Rushing was the principle architect and implementer of the ADaM data mining system. Drs. Sara Graves and Heggere Ranganath also were significant contributors to the development of ADaM. We would also like to acknowledge the valuable comments that William Thigpen and Tony Lisotta at the NASA Ames Research Center made on this paper. We also note that the CERES data used in this paper were obtained from the Atmospheric Sciences Data Center and the TMI data were obtained from the Goddard Space Flight Center's Distributed Active Archive Center. We would also like to acknowledge Ray Turney of NASA Ames Research Center for his support of our use of the Storage Resource in support of the work reported upon in this paper.

This paper was reproduced with kind permission of Springer Science and Business Media: Kluwer Academic Publishers, Journal of Grid Computing, vol. 1, issue 2, 2003, pp. 101-116, "Distributed Generation of NASA Earth Science Data Products," Bruce R. Barkstrom, Thomas H. Hinke, Shradha Gavali, Warren Smith, William J. Seufzer, Chaumin Hu, and David E. Cordner.

References

[Avery and Foster 2000] P. Avery and I. Foster, "The GriPhyN Project: Towards Petascale Virtual-Data Grids," Grid Physics Network GriPhyN 2001-14, April 17 2000.

[Barkstrom 1998] B. R. Barkstrom, "Digital Archive Issues from the Perspective of an Earth Science Data Producer", paper presented at the *Digital Archive Directions (DADS) Workshop*, June 22-26, 1998, available at http://ssdoo.gsfc.nasa.gov/nost/isoas/dads/dads21b.html.

[Barkstrom 2003] B. R. Barkstrom, "Data Product Configuration Management and Versioning in Large-Scale Production of Satellite Scientific Data," in B. Westfechtel and A. van der Hoek (eds.), SCM 2001/2003, *Lecture Notes in Computer Science* 2649, pp. 118-133, 2003.

[Baru, et al. 1998] C. Baru, R. Moore, A. Rajasekar, and M. Wan, "The SDSC Storage Resource Broker," Proceedings of the CASCON'98, Toronto, Canada, 1998.

[Chervenak, et al. 2001] A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, and S. Tuecke, "The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets," Journal of Network and Computer Applications, vol. 23, pp. 187-200, 2001.

[Cog 2003] Commodity Grid Kits, http://www-unix.globus.org/cog/.

[Data Mining Workshop 2003] Workshop on Data Mining and Exploration Middleware for Distributed and Grid Computing, University of Minnesota Supercomputing Center, September 2003.

[Du and Agrawal 2002] Du, Wei and Gagan Agrawal, "Developing Distributed Data Mining Implementations for a Grid Environment," *Proceedings 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*, Berlin, Germany, May 2002.

[Devlin 1995] K. I. Devlin, Application of the 85 GHz Ice Scattering Signature to a Global Study of Mesoscale Convective Systems. Master's thesis, Meteorology, Texas A&M University, August 1995.

[Foster, et al. 2003] I. Foster, J. Vockler, M. Wilde, and Y. Zhao, "The Virtual Data Grid: A New Model and Architecture for Data-Intensive Collaboration," Proceedings of the Conference on Innovative Data System Research, 2003.

[Ghiselli 2002] A. Ghiselli, "DataGrid Prototype 1," Proceedings of the TERENA Networking Conference, 2002.

[GridFTP 2002] W. Allcock, J. Bresnahan, I. Floster, L. Liming, J. Link, and P. Plaszczac, "GridFTP Update January 2002", Globus Project Technical Report, http://www.globus.org/datagrid/deliverables/GridFTP-Overview-200201.pdf, January 2002.

[Guy, et al. 2002] L. Guy, P. Kunszt, E. Laure, H. Stockinger, and K. Stockinger, "Replica Management in Data Grids," Proceedings of the 5th Global Grid Forum Meeting, Edinburgh, Scotland, 2002.

[Hinke 2000a] Thomas H. Hinke, John Rushing, Heggere Ranganath and Sara J. Graves, "Techniques and Experience in Mining Remotely Sensed Satellite Data," *Artificial Intelligence Review: Issues on the application of data mining,* Vol. 14, No. 6, December 2000, pp. 503-531.

[Hinke 2000b] Thomas H. Hinke, and Jason Novotny, "Data Mining on NASA's Information Power Grid," *Proceedings Ninth IEEE International Symposium on High Performance Distributed Computing,* Pittsburgh, Pennsylvania, August 2000.

[Hinke 1997a] Thomas H. Hinke, John Rushing, Shalini Kansal, Sara J. Graves, Heggere Ranganath and Evans Criswell, "Eureka Phenomena Discovery and Phenomena Mining System," *Proceedings: 13th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography and Hydrology,* Long Beach, California, February, 1997.

[Hinke 1997b] Thomas H. Hinke, John Rushing, Shalini Kansal, Sara J. Graves and Heggere Ranganath. "For Scientific Data Discovery: Why Can't the Archive be More Like the Web," *Proceedings Ninth International Conference on Scientific Database Management*, Evergreen State College, Olympia, Washington, August, 1997.

[Johnston, et al. 1999] W. Johnston, D. Gannon, and B. Nitzberg, "Grids as Production Computing Environments: The Engineering Aspects of NASA's Information Power Grid," Proceedings of the 8th IEEE International Symposium on High Performance Distributed Computing, 1999.

[Laszewski 2000] G. V. Laszewski, I. Foster, J. Gawor, W. Smith, and S. Tuecke, "CoG Kits: A Bridge between Commodity Distributed Computing and High-Performance Grids," Proceedings of the ACM Java Grande Conference, 2000.

[Mining Workshop 1999] *NASA Workshop on the Issues in the Application of Data Mining to Scientific Data*. 1999, NASA Goddard Space Flight Center, http://datamining.itsc.uah.edu/meeting/DMFinalReport.pdf.

[Mining Middleware 2003] Data Mining and Exploration Middleware for Distributed and Grid Computing, University of Minnesota Supercomputing Institute, September 18-19, 2003, http://www.msi.umn.edu/general/Symposia/dmem/agenda.htm

[NRC 2003] *Government Data Centers: Meeting Increasing Demands*. 2003, National Research Council of the National Academies, Washington, DC, http://www.nap.edu.

[NVO] "U.S. National Virtual Observatory," http://www.us-vo.org

[PPDG] "Particle Physics Data Grid," http://www.ppdg.net[Schwaller 1996] M. Schwaller, B. Krupp and W. North, Science Data Plan for the EOS Data and Information System, Technical report, Goddard Space Flight Center, July 1996.

[Wan 2003] Michael Wan, Arcot Rajasekar, Regan Moore and Phil Andrew, "A Simple Mass Storage System for the SRB Data Grid," 20th IEEE/ 11th NASA Goddard Conference on Mass Storage Systems & Technologies (MSST2003), San Diego, CA, April 2003.